[0343] Formula. Each derivation consists of two parts: (1) a collection of Attributes that are used to derive a value and (2) the formula that uses those Attribute values to generate a derived value. This column contains the formula used to generate a derived value.

[0344] In an alternative embodiment, each entry in the Derivations table 1170 defines how to calculate the value of a derived Attribute, a calculation that involves other Attribute values. The entries in the Attribute Derivations table 1780 list the Attributes required by each calculation in the Derivations table 1770. The required Attributes, in turn, are used during data maintenance to (a) identify when a derived, instantiated Attribute must be updated (because one of the Attributes required by a derivation was changed) and (b) generate the SQL statement that accumulates all of the Attribute values together in order to compute a derived value. This table consists of the following two columns.

[0345] Derived Attribute ID. Each entry in this table lists an Attribute that is required for a derivation, and the derivation is identified by the Attribute ID of the Attribute that is being derived. This column lists the ID of the derived Attribute.

[0346] Required Attribute ID. Each entry in this table lists an Attribute that is required for a derivation. This column lists the ID of a required Attribute.

[0347] A validation is a check on an Attribute value that should be implemented before changes to an Attribute value are allowed. Some validations (e.g., a value for an Attribute with a lookup table must come from the list of values in

that lookup table) are built-in to the data model and are not listed in the Attribute Validations table 1795. Those validations that are not built-in to the data model (e.g., the median-type of an undivided road must be "none") are specified in this table.

[0348] Active ([Active_Ind]). Validations may be temporarily deactivated by setting this Boolean value to False.

[0349] Description ([Descr]). Each validation may have an optional description, which describe the validation.

[0350] Name ([Name]). In order to facilitate user definition of the validations, each validation is assigned a name, which must be unique among the validations. These names are used in reports and user interfaces that list failed data updates and the reasons for the failure. This column contains the validation name.

[0351] Validation Formula ([Formula]). Each validation is implemented as a formula that computes a value of True if a validation is satisfied and a value of False if a validation fails. This column contains the validation formula.

[0352] Validation ID ([Validation_ID]). Each validation is identified by a unique numeric identifier, the Validation ID.

[0353] Validation Type ([Validation_Typ_Cd]). Failure of a validation may result in a variety of responses (e.g., abort a data update, confirm an update before proceeding). The validation type specifies what happens if a validation fails.

[0354] Attribute values can be validated by cross-checking different Attribute values. Each validation consists of (a) a formula used to perform the validation and

(b) a list of Attributes whose values are required by the formula. This table is a join table that indicates the Attributes required by a validation formula.

[0355] Attribute ID ([Attribute_ID]). Each validation consists of the Attributes required for the validation and the validation formula. The Attribute ID identifies a required Attribute for a validation.

[0356] Triggers Validation ([Trigger_Ind]). Some validations that are involved in a validation should not trigger a validation if the value of that Attribute changes. The Trigger Ind indicates whether a change in the value of the Attribute specified by the Attribute ID should trigger a validation.

[0357] Validation ID ([Validation_ID]). Each validation consists of the Attributes required for the validation and the validation formula. The validation ID identifies the row in the Validations table that contains the validation formula.

[0358] One advantage of the present system and method over the prior art is the ability to generate maps that are based on the data in the database. In many cases (e.g., accidents), the maps are generated directly from the data tables themselves. In other cases, however, the maps are generated from data that is derived from the data tables. For example, "turbo layers" are used in order to render maps more quickly, and a turbo layer is derived from the underlying mappable data through geographical unions of shapes in the underlying map data. Also, maps that are related to data that exist in more than one Application Table must rely on a derived data table. The Map Dataset table 1730 describes each of the mappable datasets. It contains the following columns: